Chris Piech and Mehran Sahami
Code in Place

Section #5
May 12, 2020

Section Handout #5: DataScience

Problems written by Chris Piech

We are going to take the first step into data science. You will learn about lists and files!

1. The Index Game

As a warmup, read this code and play the game a few times. Use this mental model of the **list**:

Value:	'Julie'	'Mehran'	'Simba'	'Ayesha'	'Karel'
Index:	0	1	2	3	4

```
def main():
    # 1. Understand how to create a list and add values
    # A list is an ordered collection of values
    names = ['Julie', 'Mehran', 'Simba', 'Ayesha']
    names.append('Karel')

    # 2. Understand how to loop over a list
    # this prints the list to the screen one value at a time
    for value in names:
        print(value)

    # 3. Understand how to look up the length of a list
    # use randint to select a valid "index"
    max_index = len(names) - 1
    index = random.randint(0, max_index)

    # 4. Understand how to get a value by its index
    # get the item at the chosen index
    correct_answer = names[index]

    # This is just like in Khansole Academy...
    # prompt the user for an answer, check if it is correct
    prompt = 'who is in index...' + str(index)+ '? '
    answer = input(prompt)
    if answer == correct_answer:
        print('Good job')
    else:
        print('Correct answer was', correct_answer)
```

2. Heads Up! It's File Reading

Our next goal is to learn how to read data from files. Loading data from a file can be useful for many final projects. Write a program that runs a console version of the phone game, Heads Up!

This is the actual app HeadsUp! In this section we focus on the algorithm.

How the game is played over Zoom:
When it is your turn, close your eyes.
A word will be displayed in the HeadsUp program.
The rest of the section will try and describe it *without saying the word*.
You have to guess the word as quickly as possible.

Milestones:
Milestone #1: First, load all of the words from the file **`words.txt`** into a list.
Milestone #2: Then, show a randomly chosen word from the list
Milestone #3: Repeat: wait for the user to hit enter, then show another word.

Helpful code:
This code reads a file line by line and prints each line. Modify it to populate a list:

```
file = open('words.txt') # file is a variable, 'words.txt' is a filename
for line in file:        # for-each loop gives lines one at a time
    print(line.strip())  # strip removes whitespace at the start or end
```

This code selects a random element from a list (in this case the one with name my_list)

```
chosen_value = random.choice(my_list) # comes with 'import random'
```

Your final project could be a game like this.

3. (Demo Problem) Covid-19 DataScience

Even though the section isn't long enough for this problem, we are including it so that you can see how the file reading and lists allow for data analysis.

This course exists as a way to give us all a means to improve ourselves during the time of Covid-19. A big component of our global response are "models", often written in Python, which predict how the disease will progress. In this problem we will take small steps towards working with that data. Much of the Covid-19 data from different countries is public.

For example we provide you with a file "italy.txt" which lists the **total number of confirmed cases** in Italy, one value per day, starting on Jan 22nd. This is real data provided by Johns Hopkins Uni.

italy.txt

```
0           # There were 0 confirmed cases up until Jan 22nd
0           # There were 0 confirmed cases up until Jan 23rd
0           # There were 0 confirmed cases up until Jan 24th
...         # the file has 109 lines!
215858      # the third to last line is cases up until May 7th
217185      # the second to last line is cases up until May 8th
218268      # There have been 218,268 cases up until May 9th
```

Challenge #1: Load all the values from the file into a list of integers.

Challenge #2: Count the number of non-zero values in the file (this is days since first case)

Challenge #3: Create a list which stores how many new cases there were each day (new cases on a given day are: total cases on that day - total cases on the previous day)

In the starter code we include similar files for all reported countries.

There are a lot of interesting questions that researchers are trying to answer. Can you tell if a country has turned the tide on the current wave of Covid-19? How can we predict the future number of cases? Can you find external data that helps explain the number of infections? These are unsolved problems, though you are welcome to explore the data using python and look for clues.

You could work on an analysis of this data for your final project.